Final Report for Contract NAS 9-11964

CALCULATION OF FREE-FALL TRAJECTORIES BASED ON NUMERICAL OPTIMIZATION TECHNIQUES

Prepared for

National Aeronautics and Space Administration

Manned Spacecraft Center

Houston, Texas

by

Texas Center for Research

Applied Mathematics and Mechanics

3100 Perry Lane

Austin, Texas 78731

INTRODUCTION

The purpose of this study has been the development of a means of computing free-fall (non-thrusting) trajectories from one specified point in the solar system to another specified point in the solar system in a given amount of time. The problem, therefore, is the determination of the initial velocity which--in combination with the initial position and time--can be numerically integrated forward in time to the specified final time to satisfy the final position requirement to within some given tolerance.

As stated above, the problem is that of solving a two-point boundary value problem for which the initial slope is unknown. Two standard methods of attack exist for solving two-point boundary value problems.

The first method is known as the initial value or shooting method. All unknown quantities at the initial time are guessed and the non-linear differential equations of motion are integrated numerically to the final time. If the final miss distance is unsatisfactory, some corrective scheme is applied to determine a new value for the guessed quantities and the process is repeated. A disadvantage of this method is that the initial values which were guessed must be "fairly close" to the correct values or the shooting method may not converge. "Fairly close" is a property of the individual problem for non-linear problems and it is often most of the work to determine the initial guesses which are close enough to converge the initial value process. An advantage of this method is that the differential equations of motion are satisfied to the tolerance of the integration for each iteration and the converged trajectory represents a true solution to the integration accuracy.

The second method of attack for two-point boundary value problems is to approximate the non-linear differential equations by an appropriate linearized set. The solution method then consists of solving a series of linear boundary value problems

which can be made to satisfy boundary conditions on every iterate. This series of linear boundary value problems then approaches the non-linear problem until the difference between the linear solution and the non-linear solution are less than some specified tolerance.

The method of this study uses parts of both boundary value problem solution techniques described above. A complete velocity history is guessed such that the corresponding position history satisfies the given boundary conditions at the appropriate times. An iterative procedure is then followed until the last guessed velocity history and the velocity history obtained from integrating the acceleration history agree to some specified tolerance everywhere along the trajectory. Convergence for this method is obtained for fairly poor initial guesses of the velocity history and terminal convergence is obtained in a quadratic manner.

DESCRIPTION OF THE METHOD

Given a two-point boundary value problem for which initial position and time and the final position and time are specified, determine the initial velocity which can be used in an initial value integration scheme to satisfy the final conditions.

The procedure to be followed is described in detail below. First, an initial guess is made of the entire velocity history. This guess is made in a manner which satisfies the boundary conditions when integrated. Designate this guessed velocity history as the control velocity history to differentiate it from an integrated velocity history obtained from the integrated acceleration. The position is obtained by integrating the control velocity history. The position dependent acceleration is integrated simultaneously to obtain an integrated velocity history. At the final time, two velocity histories are known: the control velocity history which satisfies the boundary conditions, and the integrated velocity history which satisfies the

non-linear differential equations of motion. An iteration scheme is now used to drive the control velocity history toward satisfaction of the non-linear differential equations while the integrated velocity is driven toward satisfaction of the boundary conditions. The method is considered to be converged when the control velocity history and the integrated velocity history agree to some specified tolerance everywhere along the path.

Therefore, a two-part correction scheme is applied to the old control velocity history to yield a new control velocity history. The first correction assures satisfaction of the boundary conditions with the new control velocity history and the second correction augments convergence to the integrated velocity history from the control velocity history.

DEVELOPMENT OF THE EQUATIONS

The formulation of the two-point boundary value problem with given initial and final positions and time may be stated in the following manner:

Determine $v(t_i) = v_i$ for

$$\dot{x} = v(t)$$

$$\dot{v} = F(x,t)$$

$$t_i,\ x_i \quad \text{Given}$$

$$t_f,\ x_f \quad \text{Given}$$

where
x is a 3 x 1 position matrix
v is a 3 x 1 velocity matrix
F is a 3 x 1 force matrix
t is a scalar

Guess a velocity history which satisfies the prescribed boundary conditions and designate this as $u(t)$. Also, designate the corresponding integrated velocity history as $\lambda(t)$.

Now
$$\dot{x} = u(t)$$
$$\dot{\lambda} = F(x,t)$$
$$t_i,\ x_i \quad \text{Given}$$
$$t_f,\ x_f \quad \text{Given}$$

The last set of equations will be solved on each iteration until the new guessed velocity history and the new integrated velocity history agree to some specified tolerance.

The integrated velocity history is to be changed by $\delta\lambda(t)$ to satisfy the boundary conditions on the next iterate while the control velocity history is changed by $\delta u(t)$ to agree with the integrated velocity history on the next iteration.

The following relations are desired on the $\ell + 1$ st iteration.

$$u_{\ell+1}(t) = \lambda_{\ell+1}(t)$$
$$u_{\ell+1}(t) = u_\ell(t) + \delta u_\ell(t)$$
$$\lambda_{\ell+1}(t) = \lambda_\ell(t) + \delta\lambda_\ell(t)$$

$$\delta u_\ell(t) = \delta\lambda_\ell(t) + \lambda_\ell(t) - u_\ell(t)$$

or
$$\delta u = \delta\lambda + P(\lambda - u)$$

The multiplier P in the above expression is a weighting matrix which can be preset to some constant value or included in the computation. Since P represents the partial derivative matrix of the control velocity to the integrated velocity, the secant method could be applied to each point to accelerate convergence. For this study, a constant scalar P where $P \leq 1$ was used.

Linearizing the differential equations leads to the following linear perturbation equations.

$$\delta\dot{x} = \delta u = \delta\lambda + P(\lambda - u)$$
$$\delta\dot{\lambda} = F_x \delta x$$

Assuming that δx and $\delta\lambda$ can be written as linear functions of their initial conditions (with time-dependent coefficients) results in the following relations when the fact that $\delta x_i = 0$ is taken into account.

$$\delta x = A(t)\ \delta\lambda_i + M(t)$$
$$\delta\lambda = B(t)\ \delta\lambda_i + N(t)$$

where A is a 3 x 3 matrix with $A_i = 0$

M is a 3 x 1 matrix with $M_i = 0$

B is a 3 x 3 matrix with $B_i = 0$

N is a 3 x 1 matrix with $N_i = 0$

Now, the last relations are differentiated to yield

$$\delta\dot{x} = \dot{A}\delta\lambda_i + \dot{M}$$
$$\delta\dot{\lambda} = \dot{B}\delta\lambda_i + \dot{N}$$

$$\delta\dot{x} = \delta u = \delta\lambda + P(\lambda - u)$$
$$\delta\dot{\lambda} = B\delta\lambda_i + N + P(\lambda - u)$$

$$\delta\dot{\lambda} = F_x\delta x$$
$$\delta\dot{\lambda} = F_x A\delta\lambda_i + F_x M$$

Comparing coefficients, the following differential equations for the linear perturbation mapping result.

$$\dot{A} = B$$
$$\dot{B} = F_x A$$
$$\dot{M} = N + P(\lambda - u)$$
$$\dot{N} = F_x M$$

These equations are integrated along with the non-linear equations. At the final time:

$$\delta x_f = x_f - x(t_f)$$

$$\delta x_f = A_f \delta\lambda_i + M_f$$

$$\delta\lambda_i = A_f^{-1}(x_f - x(t_f) - M_f)$$

This provides the new value for the initial velocity.

$$\delta\lambda = B\delta\lambda_i + N$$

$$\delta u = \delta\lambda + P(\lambda - u)$$

$$\delta u = B\delta\lambda_i + N + P(\lambda - u)$$

$$\delta u_i = \delta\lambda_i$$

This is the procedure to follow in making changes in the old control velocity history to begin a new iteration. The iteration procedure is stopped when the maximum value of δu anywhere along the path is less than the specified tolerance.

NUMERICAL INTEGRATION AND INTERPOLATION

In order to apply the iteration scheme in the preceeding section, it would be advantageous to employ a constant stepsize numerical integration procedure so that the corrections to the control velocity history would always occur at the tabulated points of the control velocity history. The terminal convergence properties and the integration accuracy are limited in a constant stepsize integrator--particularly if any singularities are present in the interval of interest.

Due to these limitations, a variable step Runge-Kutta-Fehlberg integrator with truncation error control was chosen for the integration package. In order to use this integrator, an interpolation scheme for the control velocity history is necessary to provide values of the control velocity between the tabulated points on each iteration. The stepsize pattern

changes from iteration to iteration so the tabulated control velocity points also change from iteration to iteration. For the interpolator, a cubic spline was chosen. The cubic spline fits a third order polynomial in time between tabulated points and provides continuous function values and continuous first and second derivative values throughout the entire control velocity history. The cubic spline was chosen for its stability, availability, and applicability.

With the cubic spline fitting a third order polynomial in each stepsize interval, the natural choice of the integrator order was determined to be the RKF 3(4) integrator which integrates a third order polynomial to the specified integration tolerance. The integrator, therefore, chooses a stepsize for a third order polynomial on integration and the spline fits a third order polynomial of interpolation to this interval. Although the combination of RKF 3(4) integrator and cubic spline should be optimum for accuracy, the combination is not the optimum for minimizing storage requirements. A higher order integrator takes fewer steps for the same integration and, therefore, requires less storage for the tabulated velocity history. The spline fit to this larger interval is not as accurate as in the preceeding combination. A compromise is, therefore, indicated--the choice of integrator to couple to the cubic spline being governed by the accuracy and storage requirements. An optimum compromise between accuracy and storage is probably the RKF 4(5) integrator coupled to the cubic spline.

For the initial value process which follows the convergence of the control velocity iteration, a high order integrator [RKF 7(8)] with a fine tolerance is used to refine the trajectory. The high order integrator is used in this part for high accuracy with few steps. Since this is an initial value process, no interpolation is necessary and the differential equations are satisfied to the integration accuracy.

APPLICATIONS

The initial test of the control velocity iteration scheme was chosen to be the generation of Apollo-type earth-moon trajectories. The initial trajectories have been generated for both coplanar and non-coplanar cases. The coplanar cases utilize initial and final points which lie in the moon's orbital plane and the non-coplanar cases have final points which do not lie in this plane. The initial point was chosen for a 100 mile departure altitude from earth and a 50 mile arrival altitude at the moon.

Beginning with the coplanar cases, a control velocity history was chosen which kept the spacecraft in the lunar orbital plane at all times. The initial guess missed the desired final point by over 2000 miles. Convergence of the control velocity iteration scheme required 9 iterations for a total of less than 20 seconds computation time on the CDC 6600. Using this initial velocity in a standard perturbation scheme (initial value method), convergence to a position miss of less than 1 inch required 3 iterations for a total of 7 additional seconds.

Choosing a non-coplanar initial velocity history guess with the same initial and final points as the case above, the control velocity iteration scheme converged back to the coplanar velocity history. Again, the perturbation scheme converged in 3 iterations. Run times for both sections were comparable to the run times of the first case.

Moving the final point 24 miles out of plane caused the third case to converge to a non-coplanar velocity distribution from the initial guess of the first case. Run times were again compatible with the first case.

The control velocity iteration scheme was next applied to the generation of interplanetary trajectories. Initially, the planets were placed in circular orbits about a fixed sun. Several trajectories of the earth-mars, earth-venus, earth-mercury types have been generated for short flight times on the order of 50 to 100 days. The short flight time

allows the use of a constant velocity for the initial guess at the velocity history. The trajectories have been generated for an initial point which is at a 300 mile altitude above the earth and a final point which is 300 miles altitude at the target planet. During the computation of these trajectories, the sun and the five inner planets were active gravitational sources. Computation time and accuracy for these trajectories were comparable to those of the earth-moon trajectories. For example, a 90-day earth-venus trajectory from 300 mile altitude to 300 mile altitude converged the control velocity iteration scheme in 5 iterations for a computation time of less than 24 seconds. The perturbation scheme again converged in 3 iterations.

A second planetary ephemeris which used 3-dimensional elliptic orbits with constant elements was the next improvement. An attempt was then made to generate an earth-saturn trajectory with an intermediate jupiter flyby. The control velocity iteration scheme converged to a trajectory from the initial point to the final point but the flyby at jupiter was not present. The initial corrections to the velocity history move the trajectory path away from the flyby local extremum to the strong non-flyby extremum. A change in the flight time might bring this trajectory back within the influence of jupiter to permit a flyby.

Several other short flight time trajectories between two planets were converged and it was observed that the 3-dimensional trajectories were more difficult to converge than the 2-dimensional trajectories.

The third planetary ephemeris to be incorporated was the J.P.L. Analytical ephemeris based upon polynomial approximations to the orbital elements of the planets. This ephemeris was chosen as the final one and a specific trajectory was chosen for computation. The specific mission was an earth to venus trajectory with a heliocentric transfer angle greater than 200°. This mission was chosen for its importance in the radioactive waste disposal project.

The trajectory end points were chosen such that the initial point was at a 100 mile altitude over a specific point on the earth and the final point was chosen to be at a 100 mile altitude over a specific point on venus. The earth launch point was chosen for compatibility with existing launch facilities and the venus final point was chosen from preliminary patched conic studies.

The initial velocity guess would ideally be the velocity distribution from the patched conic trajectory which would consist of hyperbolic planetocentric segments matched in position and velocity to elliptical heliocentric segments. Unfortunately, this system was not available. The result was that Lambert's theorem was used to generate a heliocentric elliptical segment connecting the initial and final points. If the planetary gravitational fields are introduced full strength, the trajectory integration requires more integration steps than the allotted storage permits.

The reason for this was that the elliptic section which was the solution of Lambert's theorem passed through the earth near to its center. The variable step integrator, therefore, needed more and smaller steps for an accurate integration. To circumvent this difficulty, the planet's gravitational strength was gradually increased from zero to 100% in steps. Lambert's theorem provided the solution for 0%, then a 1% problem was converged, then a 10% problem and finally a 100% problem. This allows the control velocity iteration scheme to shape the trajectory gradually. More time is required for this procedure but final convergence is obtained without exceeding the allotted storage.

Beginning with the Lambert's theorem velocity history guess, 4 iterations were required for convergence of the 1% problem, 6 iterations were required for convergence of the 10% problem, and 10 iterations were required for convergence of the 100% problem. The total run time was 179.59 seconds.

A better initial velocity guess history would do away with the relaxation of the planetary gravitational fields and also

augment convergence. This would greatly reduce the total required run time.

The initial velocity was taken from the above converged run and used in a standard initial value perturbation scheme. The integrated terminal miss was reduced somewhat but quadratic convergence was not obtained. The initial integration of this scheme usually shows a greater terminal miss than the converged velocity guess scheme since the differential equations of motion are obeyed exactly in the standard perturbation scheme. A reduce-the-norm iteration procedure was employed with the standard perturbation scheme which would not accept an iterate unless it reduced the norm of the miss distance from the last accepted iterate.

To generate a flyby trajectory, two problems of the preceeding type were solved. In the velocity guess iteration scheme, the joining point was fixed at venus and the two trajectories (earth-to-venus and venus-to-terminal-point) were matched in position (to integration accuracy) but not in velocity at the joining point. An iteration scheme was then employed in the standard perturbation scheme which was to iterate out the velocity of each segment to vary. This process was successful in decreasing the velocity mismatch to 30-40 fps but integration accuracy was such that no further reduction could be obtained.

CONCLUSIONS AND RECOMMENDATIONS

The control velocity iteration is a very powerful tool for solving two-point boundary value problems. Convergence can be obtained to the accuracy of the interpolation scheme even for a somewhat poor initial guess. The method is particularly applicable to the computation of velocity histories for interplanetary trajectories where the initial velocity guess might not converge the standard perturbation scheme.

The generation of planetary flyby trajectories by the velocity guess iteration will require the inclusion of a constraint

at the flyby planet in order to assure that the trajectory converges to the flyby which is a weak local extremum instead of the non-flyby which is a much stronger extremum. Without this constraint, the early stages of the velocity iteration scheme move the trajectory away from the flyby trajectory to a non-flyby trajectory.

The accuracy of the integration should be improved in the neighborhood of the planets if the integration scheme is switched from heliocentric ecliptic to planetocentric ecliptic when inside the planet's sphere of influence. More accuracy of the integrator and, therefore, of the interpolator should improve the convergence characteristics of the velocity guess iteration. The accuracy should also be reflected in the initial value perturbation scheme by providing much more accurate information for computing changes in the initial velocity.